

Amendments to the Claims:

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

1. (Original) A method in a data processing system for storing data in a file system, the method comprising:
determining whether space is available in an inode for a file in the file system; and
responsive to space being available, storing the data in the inode.
2. (Original) The method of claim 1 further comprising:
determining whether additional data is present; and
responsive to the additional data being present, storing the additional data in a partially filled block of another file.
3. (Original) The method of claim 1 further comprising:
responsive to spacing being unavailable, storing the additional data in a partially filled block of another file.
4. (Original) The method of claim 3, wherein the partially filled block is a last block of the another file.
5. (Original) The method of claim 1, wherein the space is located in an extension area in the inode.
6. (Original) The method of claim 1 further comprising:
determining whether a file size for the data is divisible by a block size for blocks in the file system; and
if the file size is divisible by the block size, storing the data in a block.
7. (Original) The method of claim 1 further comprising:
determining whether space is available in the inode to store the data; and
responsive to room being unavailable in the inode, storing the data in a partially filled block of another file.

8. (Original) A data processing system for storing data in a file system, the data processing system comprising:

- a bus system;
- a communications unit connected to the bus system;
- a memory connected to the bus system, wherein the memory includes a set of instructions; and
- a processing unit connected to the bus system, wherein the processing unit executes the set of instructions to determine whether space is available in an inode of the file in the file system; and store the data in the inode in response to space being available.

9. (Original) A data processing system for storing data in a file system, the data processing system comprising:

- determining means for determining whether space is available in an inode of the file in the file system; and
- storing means, responsive to space being available, for storing the data in the inode.

10. (Original) The data processing system of claim 9, wherein the determining means is a first determining means and the storing means is a first storing means and further comprising:

- second determining means for determining whether additional data is present; and
- second storing means, responsive to the additional data being present, for storing the additional data in a partially filled block of another file.

11. (Original) The data processing system of claim 9, wherein the storing means is a first storing means and further comprising:

- second storing means, responsive to spacing being unavailable, for storing the additional data in a partially filled block of another file.

12. (Original) The data processing system of claim 11, wherein the partially filled block is a last block of another file.

13. (Original) The data processing system of claim 9, wherein the space is located in an extension area in the inode.

14. (Original) The data processing system of claim 9, wherein the determining means is a first determining means and the storing means is a first storing means and further comprising:

second determining means for determining whether a file size for the data is divisible by a block size for blocks in the file system; and

second storing means, if the file size is divisible the by the block size, for storing the data in a block.

15. (Original) The data processing system of claim 9, wherein the determining means is a first determining means and the storing means is a first storing means and further comprising:

second determining means for determining whether space is available in the inode to store the data; and

second storing means, responsive to room being unavailable in the inode, for storing the data in a partially filled block of another file.

16. (Previously Presented) A recordable-type medium having a computer program product for storing data in a file system, the computer program product comprising:

first instructions for determining whether space is available in an inode of the file in the file system; and

second instructions, responsive to space being available, for storing the data in the inode.

17. (Previously Presented) The recordable-type medium of claim 16 wherein the computer program product further comprises:

third instructions for determining whether additional data is present; and

fourth instructions, responsive to the additional data being present, for storing the additional data in a partially filled block of another file.

18. (Previously Presented) The recordable-type medium of claim 16 wherein the computer program product further comprises:

third instructions, responsive to spacing being unavailable, for storing the additional data in a partially filled block of another file.

19. (Previously Presented) The recordable-type medium of claim 18, wherein the partially filled block is a last block of another file.

20. (Previously Presented) The recordable-type medium of claim 16, wherein the space is located in an extension area in the inode.

21. (Previously Presented) The recordable-type medium of claim 16 wherein the computer program product further comprises:

third instructions for determining whether a file size for the data is divisible by a block size for blocks in the file system; and

fourth instructions, if the file size is divisible the by the block size, for storing the data in a block.

22. (Previously Presented) The recordable-type medium of claim 16 wherein the computer program product further comprises:

third instructions for determining whether space is available in the inode to store the data; and

fourth instructions, responsive to room being unavailable in the inode, for storing the data in a partially filled block of another file.